- on the other hand, Gupta discloses a central transaction system that provides for transactions in which separate applications are integrated. The applications are independent and, more importantly, incompatible (e.g. column 4, line 66 – column 5, line 2). The purpose of the Gupta system is to allow these incompatible applications to provide transactional behavior. However, the transactional behavior is implemented by the server, on behalf of the client applications, since the applications themselves do NOT provide for transactional behavior (column 6, lines 53-56). Thus all actions that are related to transactional behavior are performed in the Gupta server, in particular the management of undo operations.

Therefore, McLaughlin's remote transactional servers and local transactional servers cannot be identified with Gupta's server and clients, respectively. If a transactional server as disclosed in McLaughlin must be identified with an entity disclosed in Gupta, it can only be identified with the combination of the server machine 220 and client machines 250, since only this combination exhibits transactional behavior.

Consequently, remote servers and local servers disclosed by McLaughlin cannot be matched with the clients and servers of Gupta, and reading McLaughlin on the first limitations of claim 23 and Gupta on the remaining limitations cannot be done consistently.

McLaughlin and Gupta, even when combined, do not show the features of claim 23

Independent of the previous discussion, even if for sake of discussion it were to be assumed that McLaughlin and Gupta could be combined, this would still not lead to the limitations of claim 23. This is explained in the following in more detail, following the text of claim 23 (**in boldface**):

23. A computerized data management system, referred to as transactional service, wherein the transactional service is executed on a local server, comprising:
one or more operations that can be invoked by remote clients,
> **wherein the remote clients are executed on a remote server that is distinct from the local server, and**
> **wherein some or all such remote clients have one or more associated transaction contexts;**
an invocation of the service, by a remote client, also containing partial or complete information that indicates or contains said client's transaction context or contexts:

In the preceding (the first three) claim paragraphs, it is defined that the remote clients have one or more associated transaction contexts. The present (fourth) paragraph states that the transaction context is indicated or contained in information from an invocation by a remote client. However, this is not consistent with Gupta, since Gupta states explicitly that the clients do not know anything about transactions and do not exhibit transactional behaviour:

> *"API 240 [of client machine 250] does not provide any interfaces for interaction with any external transaction coordinator. Thus it does not expose, for external use, any notion of "transactional" behavior." (Gupta, col. 6, lines 53-56)*

Therefore, information provided by the Gupta clients cannot contain partial or complete information that indicates or contains said client's transaction context.

> **an invocation of the service, by a remote client, of an operation leading to a new transaction different from, but possibly related to, any existing client transaction,**
> > **wherein such an operation-level transaction is committed before the client transaction context is terminated before globalCommit notification;**

The Office Action identifies the preceding feature as being supposedly disclosed in Gupta, col. 12, lines 28-57. However, this feature cannot be found in the referenced section of Gupta, and not even in Gupta as a whole, because committing an operation-level transaction before terminating the transaction context of the client that invoked the operation is something that

- 3 -

happens in transaction systems with two-phase commitment. This is also a direct and unequivocal consequence by the fact that the global commit feature is recited, which is an element of two-phase commitment protocol. Gupta does not mention two-phase commitment.

Furthermore, a two-phase commitment would never be an issue in the Gupta system, because the Gupta system implements a monolithic central transaction server, whereas two-phase commitment is used to solve problems of distributed transaction systems. Consequently, Gupta does not disclose an operation-level transaction being committed before the client transaction context is terminated before globalCommit notification.

(Note regarding claims 30 and 37: for the same reason, since Gupta does not disclose two-phase commitment and globalCommit, these claims related to globalCommit cannot be disclosed by Gupta).

Furthermore, McLaughlin teaches against the commitment of subtransactions. See col. 6, lines 9-29:

> *"Substituting a compound nested transaction type for a compound multilevel transaction type (1) eliminates the subtransaction commit process and containment transaction and (2) vouchsafes an all or nothing commit for the distributed transaction components [11].*

Thus, not only does Gupta not disclose the preceding feature, but the combination with McLaughlin teaches against the feature.

the transactional service locally, on the local server, maintaining an undo operation for such a committed operation; and

The Office Action identifies the preceding feature as being supposedly disclosed in Gupta, col. 6, lines 12-20. However, under such an interpretation, this would mean that the Gupta server

machine is read on the local server of the present application. This follows because only the Gupta server machine knows about transaction and maintains undo operations associated with subtransactions.

> **a failing or failed remote client transaction context leading to the execution of the locally maintained undo operations of the corresponding committed invocations in the transactional service.**

The Office Action states that the preceding feature is supposedly disclosed in Gupta, col. 7, lines 42-46. However, if the local server of the present invention is identified with Gupta's server machine, then the remote client would have to be identified with one of Gupta's clients, which is inconsistent with the present claim since - according to the claim - the remote client is transactional and Gupta's clients, by definition, are not.

CONCLUSION

For the foregoing reasons, the Applicant requests that all claim rejections be withdrawn and that a Notice of Allowance be granted.

Respectfully submitted,

/s/

_____
Carl Oppedahl
USPTO Reg. no. 32746